

This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

**As rescanning documents *will not* correct images,
please do not report the images to the
Image Problems Mailbox.**



Docket No.: R-0748

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to: Assistant Commissioner for Patents, Washington, D.C. 20231.

By: 

Date: December 6, 2001

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant : Albrecht Mayer
Appl. No. : 09/974,924
Filed : October 10, 2001
Title : Programmable Unit

CLAIM FOR PRIORITY

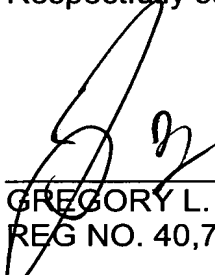
Hon. Commissioner of Patents and Trademarks,
Washington, D.C. 20231

Sir:

Claim is hereby made for a right of priority under Title 35, U.S. Code, Section 119, based upon the German Patent Application 100 50 011.0 filed October 10, 2000.

A certified copy of the above-mentioned foreign patent application is being submitted herewith.

Respectfully submitted,

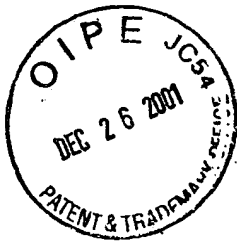


GREGORY L. MAYBACK
REG NO. 40,719

Date: December 6, 2001

Lerner and Greenberg, P.A.
Post Office Box 2480
Hollywood, FL 33022-2480
Tel: (954) 925-1100
Fax: (954) 925-1101

/mjb



**Prioritätsbescheinigung über die Einreichung
einer Patentanmeldung**

Aktenzeichen: 100 50 011.0
Anmeldetag: 10. Oktober 2000
Anmelder/Inhaber: Infineon Technologies AG,
München/DE
Bezeichnung: Programmgesteuerte Einheit
IPC: G 06 F 11/36

**Die angehefteten Stücke sind eine richtige und genaue Wiedergabe der ur-
sprünglichen Unterlagen dieser Patentanmeldung.**

München, den 04. Oktober 2001
Deutsches Patent- und Markenamt
Der Präsident
Im Auftrag

Hiebinger

Beschreibung

Programmgesteuerte Einheit

- 5 Die vorliegende Erfindung betrifft eine Vorrichtung gemäß dem Oberbegriff des Patentanspruchs 1, d.h. eine programmgesteuerte Einheit mit einer oder mehreren Programmausführungs-Einheiten zur Ausführung eines Programms, wobei wenigstens einer der Programmausführungs-Einheiten eine
- 10 Anhalte-Einrichtung zugeordnet ist, durch welche die Programmausführung durch die Programmausführungs-Einheit, der die Anhalte-Einrichtung zugeordnet ist, angehalten werden kann.
- 15 Programmgesteuerte Einheiten dieser Art sind beispielsweise Mikroprozessoren, Mikrocontroller, Signalprozessoren etc.

- Die Anhalte-Einrichtung, durch welche die Programmausführung durch eine ein Programm ausführende Programmausführungs-
- 20 Einheit der programmgesteuerten Einheit angehalten werden kann, ist beispielsweise ein sogenanntes On-Chip-Debug-Support- bzw. OCDS-Modul.

- OCDS-Module besitzen die Fähigkeit, das Auftreten bestimmter Ereignisse (Breakpoints), beispielsweise den Zugriff auf eine bestimmte Adresse im Programmspeicher oder im Datenspeicher, die Ausführung eines bestimmten Befehls, das Auslesen oder
- 30 Einschreiben bestimmter Daten aus bzw. in einen Speicher, zu überwachen, und im Ansprechen auf das Eintreten des Ereignisses die Programmausführung durch die Programmausführungs-Einheit, welcher sie zugeordnet ist, anzuhalten.

- Wenn die Programmausführung angehalten ist, kann ein externer Debugger die angehaltene programmgesteuerte Einheit und/oder
- 35 das diese enthaltende System nach möglichen Ursachen für eine nicht ordnungsgemäße Funktion untersuchen und/oder - bei-

spielsweise durch die Veränderung von Registerinhalten - in einen bestimmten Zustand versetzen.

Insbesondere, aber nicht ausschließlich im letztgenannten
5 Fall, läßt man die programmgesteuerte Einheit danach wieder weiterlaufen, um die Fehlersuche fortzusetzen oder die Auswirkungen der vorgenommenen Veränderungen zu überprüfen.

Aufgrund des immer komplexer werdenden Aufbaus der programm-
10 gesteuerten Einheiten und der diese enthaltenden Systeme kommt es jedoch bisweilen vor, daß die programmgesteuerte Einheit auch dann, wenn während des Anhaltens der Programmausführung keine bleibenden Veränderungen vorgenommen wurden, bei der Fortsetzung der Programmausführung nicht so weiter-
15 läuft wie sie es getan hätte, wenn die Programmausführung nicht angehalten worden wäre. Die Folge ist, daß in solchen programmgesteuerten Einheiten oder Systemen auftretende Fehler nicht oder nur bedingt erkannt und analysiert werden können.

20

Der vorliegenden Erfindung liegt daher die Aufgabe zugrunde, die programmgesteuerte Einheit gemäß dem Oberbegriff des Patentanspruchs 1 derart weiterzubilden, daß diese zumindest dann, wenn in ihr während des Anhaltens der Programmausführung keine oder jedenfalls keine nicht wieder rückgängig gemachten Manipulationen erfolgt sind, bei der Fortsetzung der Programmausführung unter allen Umständen genau so weiterläuft wie sie es getan hätte, wenn die Programmausführung nicht angehalten worden wäre.

30

Diese Aufgabe wird erfindungsgemäß durch die in Patentanspruch 1 beanspruchte programmgesteuerte Einheit gelöst.

Die erfindungsgemäßen programmgesteuerten Einheiten zeichnen
35 sich dadurch aus, daß die Anhalte-Einrichtung veranlassen kann, daß neben der Programmausführungs-Einheit, welcher sie

zugeordnet ist, auch andere Komponenten der programmgesteuerten Einheit angehalten werden.

5 Dadurch, daß im Falle eines Anhaltens der Programmausführung nicht nur die das Programm ausführende Programmausführungseinheit, sondern auch weitere Komponenten der programmgesteuerten Einheit, ins besondere weitere Programmausführungseinheiten und/oder mit den anzuhaltenden Programmausführungseinheiten kooperierende Komponenten der programmgesteuerten
10 Einheit angehalten werden, kann erreicht werden, daß sich während des Anhaltens der Programmausführungseinheit nichts verändert, das Einfluß auf eine spätere Fortsetzung der Programmausführung und/oder die dabei erhaltenen Ergebnisse hat oder haben könnte.

15 Sofern während des Anhaltens der Programmausführung nicht absichtlich bleibende Veränderungen in der programmgesteuerten Einheit vorgenommen wurden, kann eine wie beansprucht ausgebildete programmgesteuerte Einheit nach dem Anhalten der
20 Programmausführung zuverlässig so weiterlaufen, wie sie es getan hätte, wenn sie nicht angehalten worden wäre.

Vorteilhafte Weiterbildungen der Erfindung sind den Unteransprüchen, der folgenden Beschreibung und der Figur entnehmbar.

Die Erfindung wird nachfolgend anhand eines Ausführungsbeispiels unter Bezugnahme auf die Figur näher beschrieben.

30 Die Figur zeigt schematisch den Aufbau der nachfolgend näher beschriebenen programmgesteuerten Einheit.

Bei der programmgesteuerten Einheit handelt es sich im betrachteten Beispiel um einen Mikrocontroller. Es sei jedoch
35 bereits an dieser Stelle darauf hingewiesen, daß hierauf keine Einschränkung besteht; die nachfolgend beschriebenen

Besonderheiten der programmgesteuerten Einheit können auch bei beliebigen anderen programmgesteuerten Einsatz kommen.

Der Vollständigkeit halber sei darauf hingewiesen, daß von der betrachteten programmgesteuerten Einheit nur die vorliegend besonders interessierenden Bestandteile derselben gezeigt sind und beschrieben werden.

Wie aus der Figur ersichtlich ist, umfaßt die betrachtete programmgesteuerte Einheit

- eine erste Programmausführungs-Einheit CORE1,
- ein der ersten Programmausführungs-Einheit CORE1 zugeordnetes On-Chip-Debug-Support- bzw. OCDS-Modul OCDS1,
- eine der ersten Programmausführungs-Einheit CORE1 zugeordnete (Master-)Schnittstelle MIF1 zum Anschluß an einen Bus BUS,
- eine zweite Programmausführungs-Einheit CORE2,
- ein der zweiten Programmausführungs-Einheit CORE2 zugeordnetes On-Chip-Debug-Support- bzw. OCDS-Modul OCDS2,
- eine der zweiten Programmausführungs-Einheit CORE2 zugeordnete (Master-)Schnittstelle MIF2 zum Anschluß an den bereits erwähnten Bus BUS,
- eine erste Peripherie-Einheit P1,
- eine der ersten Peripherie-Einheit P1 zugeordnete (Slave-) Schnittstelle SIF1 zum Anschluß an den bereits erwähnten Bus BUS,
- eine zweite Peripherie-Einheit P2,
- eine der zweiten Peripherie-Einheit P2 zugeordnete (Slave-) Schnittstelle SIF2 zum Anschluß an bereits erwähnten Bus BUS,
- den bereits mehrfach erwähnten Bus BUS, wobei dieser Bus aus einem ersten Bus-Teil BUS1, einem zweiten Bus-Teil BUS2, und einer die Bus-Teile BUS1 und BUS2 miteinander verbindenden Bus Bridge BB,
- eine Anhaltekonfigurationsvorrichtung CONF, und
- eine Anhalteverzögerungsvorrichtung DEL.

Die erste Programmausführungs-Einheit CORE1 und die zweite Programmausführungs-Einheit CORE2 führen unabhängig voneinander das selbe oder verschiedene Programme aus.

- 5 Die Peripherie-Einheiten P1 und P2 sind beispielsweise A/D-Wandler, Timer, DMA-Controller oder sonstige Einheiten, die in programmgesteuerten Einheiten neben einer oder mehreren Programmausführungs-Einheiten zum Einsatz kommen können und mit diesen kooperieren.

10

Die Bus Bridge BB dient zur Verbindung der Bus-Teile BUS1 und BUS2. Bus Bridges werden vorgesehen, damit die Datenübertragung störenden Leitungskapazitäten gering gehalten werden können und/oder damit unterschiedlich schnelle und/oder

- 15 unterschiedlich aufgebaute oder arbeitende Busse bzw. Bus-Teile aneinander gekoppelt werden können. Die Bus Bridge BB nimmt vom Bus-Teil BUS1 Daten entgegen und leitet sie an den Bus-Teil BUS2 weiter, und nimmt umgekehrt vom Bus-Teil BUS2 Daten entgegen und leitet sie an den Bus-Teil BUS1 weiter;
20 die weiterzuleitenden Daten werden normalerweise in einem Puffer zwischengespeichert.

Die den Programmausführungs-Einheiten zugeordneten OCDS-Module OCDS1 und OCDS2 ermöglichen es, die Programmausführungs-Einheiten, denen sie zugeordnet sind, durch einen externen Debugger zu debuggen.

Die vorliegend besonders interessierende Eigenschaft der OCDS-Module OCDS1 und OCDS2 besteht darin, daß sie als

- 30 Anhalte-Einrichtung zum Anhalten der Programmausführung durch die Programmausführungs-Einheit, welcher sie zugeordnet sind, verwendet werden können; die OCDS-Module besitzen die Fähigkeit, das Auftreten bestimmter Ereignisse (Breakpoints) in den Programmausführungs-Einheiten, denen sie zugeordnet sind,
35 beispielsweise den Zugriff auf eine bestimmte Adresse im Programmspeicher oder im Datenspeicher, die Ausführung eines bestimmten Befehls, das Auslesen oder Einschreiben bestimmter

Daten aus bzw. in einen Speicher zu überwachen, und im Ansprechen auf das Eintreten des Ereignisses die Programmausführung durch die Programmausführungs-Einheit, welcher sie zugeordnet sind, anzuhalten. Insoweit besteht kein Unterschied zu herkömmlichen OCDS-Modulen.

Die im betrachteten Beispiel durch die OCDS-Module OCDS1 und OCDS2 gebildeten Anhalte-Einrichtungen weisen darüber hinaus die Eigenschaft auf, daß sie veranlassen können, daß neben der Programmausführungs-Einheit, welcher sie jeweils zugeordnet sind, auch andere Komponenten der programmgesteuerten Einheit angehalten werden.

Die anderen Komponenten umfassen im betrachteten Beispiel

- eine oder mehrere Programmausführungs-Einheiten, welchen keine oder eine andere als die das Anhalten der Programmausführung initiiierende Anhalte-Einrichtung zugeordnet ist, und
- eine oder mehrere mit den Programmausführungs-Einheiten kooperierende Einheiten der programmgesteuerten Einheit.

Im betrachteten Beispiel

- kann das erste OCDS-Modul OCDS1 veranlassen, daß neben der ersten Programmausführungs-Einheit CORE1, welcher es zugeordnet ist, auch die zweite Programmausführungs-Einheit CORE2 sowie die erste Peripherie-Einheit P1 und die zweite Peripherie-Einheit P2 angehalten werden, und
- kann das zweite OCDS-Modul OCDS2 veranlassen, daß neben der zweiten Programmausführungs-Einheit CORE2, welcher es zugeordnet ist, auch die erste Programmausführungs-Einheit CORE1 angehalten wird.

Der Auslöser dafür, daß das zweite OCDS-Modul OCDS2 das Anhalten der zweiten Programmausführungs-Einheit CORE2 und der ersten Programmausführungs-Einheit CORE1 veranlaßt, besteht darin, daß diese die Erfüllung einer Breakpoint-Bedingung
 5 durch die zweite Programmausführungs-Einheit CORE2 feststellt.

Der Auslöser dafür, daß das erste OCDS-Modul OCDS1 das Anhalten der ersten Programmausführungs-Einheit CORE1, der zweiten
 10 Programmausführungs-Einheit CORE2, und der Peripherie-Einheiten P1 und P2 veranlaßt, besteht darin, daß das erste OCDS-Modul OCDS1 die Erfüllung einer Breakpoint-Bedingung durch die erste Programmausführungs-Einheit CORE1 feststellt. Es kann (muß aber nicht) vorgesehen werden, daß das erste OCDS-
 15 Modul OCDS1 auch im Fall, daß es vom zweiten OCDS-Modul OCDS2 dazu veranlaßt wird, die erste Programmausführungs-Einheit CORE1 anzuhalten, das Anhalten der ersten Peripherie-Einheit P1 und der zweiten Peripherie-Einheit P2 veranlaßt; ob dies der Fall ist, ist im ersten OCDS-Modul OCDS1 vorzugsweise
 20 einstellbar.

Das Anhalten der zweiten Programmausführungs-Einheit CORE2 durch das erste OCDS-Modul OCDS1 erfolgt durch ein Signal BRK1, welches das erste OCDS-Modul OCDS1 an das zweite OCDS-
 Modul OCDS2 ausgibt, und welches das zweite OCDS-Modul OCDS2 zum Anlaß nimmt, die zweite Programmausführungs-Einheit CORE2 anzuhalten; das Anhalten der ersten Programmausführungs-
 Einheit CORE1 durch das zweite OCDS-Modul OCDS2 erfolgt durch ein Signal BRK2, welches das zweite OCDS-Modul OCDS2 an das
 30 erste OCDS-Modul OCDS1 ausgibt, und welches das erste OCDS-Modul OCDS1 zum Anlaß nimmt, die erste Programmausführungs-Einheit CORE1 anzuhalten. Der Vollständigkeit halber sei angemerkt, daß die Signale BRK1 und BRK2 auch direkt der Programmausführungs-Einheit zugeführt werden kann, deren Anhalten
 35 veranlaßt werden soll; dadurch können auch Programmausführungs-Einheiten angehalten werden, denen kein OCDS-Modul zugeordnet ist.

Die Signale BRK1 und BRK2 passieren auf ihrem Weg zu den OCDS-Modulen, für welche sie bestimmt sind, die Anhaltekonfigurationsvorrichtung CONF. Die Anhaltekonfigurations-
 5 vorrichtung CONF enthält Schaltelemente, über welche einstellbar ist, ob das Signal BRK1 zum zweiten OCDS-Modul OCDS2 durchgelassen wird, und/oder ob das Signal BRK2 zum ersten OCDS-Modul OCDS1 durchgelassen wird.

- 10 Das Anhalten der Peripherie-Einheiten P1 und P2 wird vom ersten OCDS-Modul OCDS1 durch die Ausgabe eines Signals SUSPEND = 1 veranlaßt. Dieses Suspend-Signal wird etwa zu dem Zeitpunkt, zu dem die Programmausführungs-Einheiten CORE1 und/oder CORE2 angehalten werden, ausgegeben, und der
 15 Anhalteverzögerungsvorrichtung DEL zugeführt. Die Anhalteverzögerungsvorrichtung DEL gibt dieses Signal mit einer bestimmten Verzögerung an die Peripherie-Einheiten P1 und P2 weiter und hält diese dadurch an; durch die verzögerte Weiterleitung des SUSPEND-Signals an die Peripherie-Einheiten P1
 20 und P2 werden diese erst eine gewisse Zeit nach dem Zeitpunkt des Anhaltens der Programmausführungs-Einheiten angehalten.

Aus dem verzögerten Anhalten der Peripherie-Einheiten P1 und P2 ergeben sich einige Vorteile, auf welche später noch genauer eingegangen wird.

- Die Weiterleitung des SUSPEND-Signals an die Peripherie-Einheiten P1 und P2 erfolgt im betrachteten Beispiel dann, wenn feststeht, daß die Komponenten der programmgesteuerten
 30 Einheiten, die Bus-Master sein können, und die Bus Bridge BB keine Daten mehr auszugeben haben und/oder nicht mehr auf bereits angeforderte oder noch anzufordernde Daten warten.

- Die Komponenten, die Bus-Master sein können, sind im betrachteten Beispiel die Programmausführungs-Einheiten CORE1 und
 35 CORE2; die restlichen Einheiten der am Bus BUS angeschlossenen Einheiten (die Peripherie-Einheiten P1 und P2) können nur

Slaves sein. Daher werden im betrachteten Beispiel die erste Programmausführungs-Einheit CORE1, die zweite Programmausführungs-Einheit CORE2, und die Bus Bridge BB daraufhin überprüft, ob sie noch Daten auszugeben haben und/oder auf bereits angeforderte oder noch anzufordernde Daten warten.

Diese Überprüfung erfolgt anhand eines BUSY-Signals, das die erste Programmausführungs-Einheit CORE1 (und/oder die dieser zugeordnete Schnittstelle MIF1), die zweite Programmausführungs-Einheit CORE2 (und/oder die dieser zugeordnete Schnittstelle MIF2), und die Bus Bridge BB ausgeben. Durch dieses BUSY-Signal zeigen die genannten Einheiten an, ob sie noch Daten auszugeben haben und/oder auf bereits angeforderte oder noch anzufordernde Daten warten. Die BUSY-Signale haben im betrachteten Beispiel jeweils

- den Wert 1, wenn die jeweilige Einheit noch Daten auszugeben hat und/oder auf bereits angeforderte oder noch anzufordernde Daten wartet, und

20

- den Wert 0, wenn die jeweilige Einheit keine Daten auszugeben hat und/oder nicht mehr auf bereits angeforderte oder noch anzufordernde Daten wartet.

Die BUSY-Signale der ersten Programmausführungs-Einheit CORE1 (und/oder der dieser zugeordneten Schnittstelle MIF1), der zweiten Programmausführungs-Einheit CORE2 (und/oder der dieser zugeordneten Schnittstelle MIF2), und der Bus Bridge BB werden durch ein ODER-Gatter OR einer ODER-Verknüpfung unterzogen; das Ausgangssignal des ODER-Gliedes OR wird der Anhalteverzögerungsvorrichtung DEL zugeführt.

Die Anhalteverzögerungsvorrichtung DEL enthält einen ersten Inverter INV1, einen zweiten Inverter INV2, ein UND-Gatter AND, und ein Flip-Flop FF, wobei

- der erste Inverter INV1 das Ausgangssignal des ODER-Gatters OR invertiert,
- 5 - der zweite Inverter INV2 das der Anhalteverzögerungsvorrichtung DEL zugeführte SUSPEND-Signal invertiert,
- das UND-Gatter AND das Ausgangssignal des ersten Inverters INV1 und das der Anhalteverzögerungsvorrichtung DEL zugeführte (nicht invertierte) SUSPEND-Signal einer UND-Verknüpfung unterzieht, und
- 10 - das Flip-Flop FF durch das Ausgangssignal des UND-Gatters AND gesetzt, durch das Ausgangssignal des zweiten Inverters INV2 zurückgesetzt wird, und das die Peripherie-Einheiten P1 und P2 anhaltende SUSPEND-Signal an diese ausgibt.
- 15

Wie leicht nachzuvollziehen ist, gibt die Anhalteverzögerungsvorrichtung DEL erst dann ein die Peripherie-Einheiten P1 und P2 anhaltendes SUSPEND-Signal ($SUSPEND = 1$) an diese aus, wenn

20

- ihr ein Signal $SUSPEND = 1$ zugeführt wird, und
- 5 - alle Signale BUSY der Programmausführungseinheiten CORE1 und CORE2 (und/oder der diesen zugeordneten Schnittstellen MIF1 und MIF2) sowie der Bus Bridge BB den Wert 0 aufweisen.

Das verzögerte Anhalten der Peripherie-Einheiten P1 und P2

30 gibt den Programmausführungseinheiten CORE1 und CORE2 oder sonstigen Bus-Mastern die Gelegenheit, die zum Zeitpunkt des Anhaltens der Programmausführung bereits begonnenen oder bestimmte andere Schreib- oder Lesezugriffe auszuführen (beispielsweise den sogenannten Write-Back-Puffer der Programmausführungseinheiten zu leeren). Dies erweist sich als vorteilhaft,

35

- weil sich dadurch das an das Anhalten der programmgesteuerten Einheit anschließende Debuggen derselben vereinfacht und zuverlässigere Ergebnisse über den aktuellen Zustand der programmgesteuerten Einheit liefert, und

5

- weil die programmgesteuerte Einheit dadurch in einen Zustand versetzt wird, durch den gewährleistet ist, daß die programmgesteuerte Einheit nach der Aufhebung des Anhalte-Zustandes ordnungsgemäß weiterarbeitet (so weiterarbeitet, wie sie es tun würde, wenn sie nicht angehalten worden wäre).

10

Wenn die programmgesteuerte Einheit nach dem Anhalten und Debuggen derselben wieder weiterlaufen soll, wird

15

- zunächst das der Anhalteverzögerungsvorrichtung DEL zugeführte Signal SUSPEND auf den Wert 0 gesetzt, was zur Folge hat daß das Flip-Flop FF zurückgesetzt wird und ein die Peripherie-Einheiten P1 und P2 nicht mehr anhaltendes Signal SUSPEND = 0 aus der Anhalteverzögerungsvorrichtung DEL ausgegeben wird, und

20

- danach das Anhalten der Programmausführung durch die Programmausführungs-Einheiten aufgehoben wird.

25

Der Vollständigkeit halber sei angemerkt, daß es hierzu nicht, oder jedenfalls nicht in allen Fällen erforderlich ist, jeweils alle Programmausführungs-Einheiten anzuhalten. Ebenso ist es hierzu nicht, oder jedenfalls nicht in allen Fällen erforderlich, jeweils alle Peripherie-Einheiten anzuhalten. Allerdings sollten Komponenten der programm-

gesteuerten Einheit, bei welchen nicht ausgeschlossen werden kann, daß sie auf eine angehaltene Komponente zugreifen, ebenfalls angehalten werden.

30

35

Es dürfte einleuchten, daß von den vorstehend beschriebenen Besonderheiten der betrachteten programmgesteuerten Einheit -

Zusammenfassung

Programmgesteuerte Einheit

- 5 Es wird eine programmgesteuerte Einheit mit einer oder mehreren Programmausführungs-Einheiten zur Ausführung eines Programms beschrieben, wobei wenigstens einer der Programmausführungs-Einheiten eine Anhalte-Einrichtung zugeordnet ist, durch welche die Programmausführung durch die Programmausführungs-Einheit, der die Anhalte-Einrichtung zugeordnet ist, angehalten werden kann. Die beschriebene programmgesteuerte Einheit zeichnet sich dadurch aus, daß die Anhalte-Einrichtung veranlassen kann, daß neben der Programmausführungs-Einheit, welcher sie zugeordnet ist, auch
- 10 andere Komponenten der programmgesteuerten Einheit angehalten werden.
- 15

